DOCUMENT RESUME

ED 267 619 FL 015 571

AUTHOR Canale, Michael; And Others
TITLE Microcomputer Software for Language Arts: Survey and Analysis. Informal Series: 63.
INSTITUTION Ontario Inst. for Studies in Education, Toronto.
REPORT NO ISBN-0-7744-5075-4
PUB DATE 85
NOTE 49p.
AVAILABLE FROM Ontario Institute for Studies in Education, 252 Bloor Street West, Toronto, Ontario M5S 1V6 Canada.
PUB TYPE Information Analyses (070) -- Reports - Research/Technical (143)

EDRS PRICE MF01 Plus Postage. PC Not Available from EDRS.
DESCRIPTORS *Classroom Techniques; *Computer Assisted Instruction; *Computer Software; *Courseware; Elementary Secondary Education; Foreign Countries; Information Systems; *Material Development; *Microcomputers; Networks; Second Language Instruction; State Surveys; Teacher Developed Materials
IDENTIFIERS Ontario

ABSTRACT

A report on a two-year study of the use of microcomputers in language arts programs in Ontario provides information on the following topics: background and orientation of the study; project goals and methods; summary of needs and recommendations from the first year of the study; results from the second year; and general recommendations. The study surveyed 79 boards of education in Ontario, analyzed the quality and suitability of available software as well as software that was being developed, and made provisions for dissemination of findings to aid future software development. The four major recommendations are that: (1) language educators reflect on not only what is being done with computers but also what is being done without them, to measure the value of instructional methods; (2) teachers give priority to software "tools" and student-oriented programs, even those not explicitly designed for language instruction, rather than teacher-oriented software whose principal advantage is convenience; (3) software analysis be viewed as an ongoing part of software use; and (4) microcomputer resource centers be established at the boards of education and used for sharing information about current microcomputer uses, computer use planning, and as a software library and electronic bulletin board. (MSE)

****
* Reproductions supplied by EDRS are the best that can be made *
* from the original document. *
****

ED267619

Microcomputer Software for Language Arts: Survey and Analysis

Michael Canale
Graham Barker
Monique Bélanger
Kathy MacRury
Robert S. McLean
Ronald G. Ragsdale

U.S. DEPARTMENT OF EDUCATION
NATIONAL INSTITUTE OF EDUCATION
EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)

X This document has been reproduced as received from the person or organization originating it.

☐ Minor changes have been made to improve reproduction quality.

• Points of view or opinions stated in this document do not necessarily represent official NIE position or policy.

"PERMISSION TO REPRODUCE THIS MATERIAL IN MICROFICHE ONLY HAS BEEN GRANTED BY

Pat Neylan,
Promotion Officer

TO THE EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)."

Informal Series/63
OISE Press/The Ontario Institute for Studies in Education

FL015571

2/3

The Ontario Institute for Studies in Education has three prime functions: to conduct programs of graduate study in education, to undertake research in education, and to assist in the implementation of the findings of educational studies. The Institute is a college chartered by an Act of the Ontario Legislature in 1965. It is affiliated with the University of Toronto for graduate studies purposes.

The publications program of the Institute has been established to make available information and materials arising from studies in education, to foster the spirit of critical inquiry, and to provide a forum for the exchange of ideas about education. The opinions expressed should be viewed as those of the contributors.

© The Ontario Institute for Studies in Education 1985
252 Bloor Street West
Toronto, Ontario
M5S 1V6

All rights reserved. No part of this publication may be reproduced in any form without permission from the publisher, except for brief passages quoted for review purposes.

Canadian Cataloguing in Publication Data

Main entry under title:
Microcomputer software for language arts

(Informal series ; 63)
Bibliography: p.
ISBN 0-7744-5075-4

1. Language arts — Ontario — Data processing.
I. Canale, Michael. 1949- II. Ontario Institute for Studies in Education. III. Series: Informal series (Ontario Institute for Studies in Education) ; 63

LB1576.M53 1985 428'.0028'5 C86-093046-7

ISBN 0-7744-5075-4 Printed in Canada

1 2 3 4 5 XX 98 88 78 68 58

Contents

Preface / v

Acknowledgements / vii

1. Background and Orientation / 1
2. Project Goals and Methods / 3
3. Year 1 Survey Results: Summary of Needs and Recommendations / 5
4. Year 2 Results: Solutions for Today — Exploiting Available Resources / 7

   Guiding Images of Language, Language Learning, and Microcomputers / 7

   Hardware: 32 Students, One Teacher, and One Machine / 9

   Exploiting Available Software / 10
5. Recommendations and Concluding Comments / 18

   Summary of Major Recommendations / 18

   Concluding Comments / 20

References / 21

Appendix A / 24

Appendix B / 35

Preface

This report presents the results of the two-year Ontario Institute for Studies in Education project "Microcomputer Software for Language Arts: Survey and Analysis." Work on this project was carried out in Boards of Education across Ontario by staff in the Franco-Ontarian Centre and in the Department of Measurement, Evaluation, and Computer Applications at OISE from May, 1982 through April, 1984.

The original motivation for this project was the lack of high quality educational software available for use in *français* (French as a first language) classrooms in Ontario's French-language schools. However, the scope of our work quickly broadened for two reasons: the small amount of French-language software available, and the large number of demands from a variety of other educators for guidance in the effective use of microcomputers in language arts programs. The scope of this project, then, has included microcomputer use in four broad contexts: English and French as first and second languages in Ontario schools. Throughout this report the word "software" is used generically to include all types of programs, e.g., courseware, lessonware, and utilities.

Our main finding is that language teachers with outstanding expertise and creativity are using the microcomputer in exemplary ways with their students at all grade levels across Ontario. This report is primarily about how these teachers and their students are able to exploit the microcomputer for language learning. It is intended both as an analysis of these exemplary uses and as a tribute to the dedicated educators who have been willing to share them with us. Based on our observations and analyses, we make a variety of recommendations throughout the report and summarize four major recommendations in the concluding section. We have also appended both the software evaluation form we helped to develop for analyzing language arts software and the mainly Canadian sources of the software that we have examined.

Acknowledgements

This research project was funded by the Ministry of Education, Ontario, under its Block Transfer Grant to the Ontario Institute for Studies in Education.

The project staff wish to thank the directors of education and contact people in the participating Boards of Education for their helpful collaboration in this project.

We are especially grateful to the many teachers, co-ordinators, consultants and principals who kindly gave freely of their time so that we could learn from their experience as innovators in implementing microcomputers in the school curricula. We also appreciated the opportunity to share our developing perspectives with them. We would particularly like to thank the following people:

From Boards of Education:

Carleton	Kerry Chalmers	Graham Fawcett
Etobicoke	Ron Hall	Penny McCahill
	Gordon Smith	
Metropolitan SSB	Guy Léger	Paul O'Connor
North York	Gary Dominick	Bart Hancock
	Susan LeRoy	Heather MacDonald
	David McGhee	Claire Piché
	John Porter	Peter Skillen
	Jim Treliving	Leslie Wasalynski
	Bill Whyte	
CSECR d'Ottawa	Yvon Lalonde	Richard Landry
	Serge Brochu	Rhéal Sabourin
Ottawa	Peter Fergus	John Fitchett
	Pat Holloway	Peter Ullrich
	Gerry Vullings	
Peel	Georgette Bolger	Gord Hamilton
	Yvette Nehou	Jo Phoenix
	Sharon Thorenton	Joanne Wilton

Scarborough	Neil Anderson	Richard Bacon
	Pat Hamilton	Doug Inkpen
	Howard Laderman	Bill Peel
	Don Rickard	François St. Pierre
	Art Willer	Trina Wood
Sudbury	Clément Dumas	Michael O'Hagan
	Dan St. Jean	
Toronto	Freda Appleyard	Peter Billingsley
	Dick Lee	Al Teliatnik
	Don Whitewood	Rick Wilkinson

From Other Organizations:

Centre Franco-Ontarien de Ressources Pédagogiques	Lise Drouin Donald George	Fraçois Dupont
Collège Algonquin	Robert Villeneuve	
Computers in Education	Rosaire Cloutier	Luella Egerton
Centre, Ontario Ministry of Education	Fran Moscall	Jerry George
Faculty of Education, University of Toronto		Joan Laird
Faculty of Education, University of British Columbia		Bernard Mohan

We also thank Ann Ovenden for her assistance with the classroom observations and Monique Nicole and Claudette Boisvert, of the Centre for Franco-Ontarian Studies at OISE, for their secretarial assistance throughout the project.

We produced the original version of this report on an Apple IIe with a "French-Canadian" keyboard, 80-column card, an Apple Dot Matrix Printer and the Skyman Interface Card. We enjoyed using the Applewriter II word processing package. We also appreciate the technical assistance of Keith Falkner of SES Computing, Toronto.

The present version is slightly revised from the mimeographed edition of June, 1984. The list of references and software sources have been updated as of October, 1985.

1. Background and Orientation

When our original funding proposal was prepared in September, 1981, it highlighted nine problems that confronted language teachers considering the use of microcomputers in their classrooms.

1. The lack of a central information bank and software distribution system.
2. The high cost of currently available software.
3. The high cost of software development.
4. The lack of both computer-literacy and programming skills among teachers and students.
5. The relatively poor generalizability of certain software from one educational context to another.
6. The questionable compatibility and validity of available software with respect to current trends in first and second language pedagogy.
7. The difficulty in adapting available software to specific student needs.
8. The questionable suitability of microcomputer-assisted learning in each of the different language skills and knowledge areas (e.g., speaking and listening).
9. Lack of feedback on the effectiveness of microcomputer use in language arts programs.

In sum, these specific problems raised four general concerns about language arts in software:

- practicality
- educational and technical quality
- effectiveness
- coordinated guidance in software development, use, and evaluation.

Much has changed in the fast-paced worlds of computers and education since this project began. Some of the problems and concerns outlined above

have begun to be addressed, others remain to be dealt with, and several new ones have emerged. During this study, we have attempted to consider a broad range of perspectives and have fluctuated widely in our attitudes and opinions from curious pessimism through cautious enthusiasm. We like to think that we now hold a blend of "sceptical optimism" concerning the practical possibilities of successfully exploiting microcomputers for language learning.

Perhaps one of the most significant positive signs has been the marked increase in the number of views expressed by language teachers and other educators strongly supporting the notions of "language and computers across the curriculum." Although the title of this project has highlighted "language arts," we have always interpreted this term broadly. We have not restricted our inquiry solely to software which focusses on language or purports to teach the sub-skills of language — in fact, we have held a certain bias against this narrow view of language learning and uses of the microcomputer. On the contrary, most of our efforts have been directed to learning about ways of using the microcomputer as a tool or medium for natural, purposeful language use. And while our focus has been necessarily limited to the learning of French and English, we are aware of the rapidly growing interest in the use of microcomputers in various foreign language programs, in special education programs, in Native language programs (Amerindian and Inuit), and in other heritage language learning as well. There are many challenges ahead.

2. Project Goals and Methods

The goal of the first year of the project was to survey current practice, development, and problems concerning software for use in language arts programs in Ontario. To this end, we developed a questionnaire, identified key contact persons in Boards of Education, and analyzed responses from 79 Boards of Education in the province. The results of this survey were published in the project's interim report released in May 1983 (Barker, MacRury, Canale, McLean, and Ragsdale 1983). The highlights are summarized in Section 3 of the present report.

During its second year, the project has addressed three further goals:

- to analyze the quality and suitability of available software widely used for language learning purposes
- to analyze relevant software under development, where feasible
- to disseminate findings and recommendations for the use and improvement of current software and the development of further software.

These objectives have been addressed through four complementary approaches:

1. Observation of experienced teachers and consultants. Based on contacts made through the survey, we identified individual teachers and language arts and computer consultants in several Boards of Education in the province who seemed to have made great strides in implementing software for language learning. They kindly agreed to let us observe in their classrooms and to share their views with us. These innovative people have provided us with a wealth of information and opinion which is reflected and interpreted throughout this book.

2. Use and analysis of software packages. A carefully selected set of software was obtained, used, and analyzed by project staff in cooperation with participating educators.

3. Ongoing review of the literature . Primarily, our focus in this activity was to develop an understanding of alternative uses of microcomputers

for language learning and, secondarily, to find out about promising software either available or under development.

4. Development of a software evaluation form. Background research was carried out related to curriculum materials evaluation and specifically software evaluation. An evaluation form specifically oriented toward materials for computer-assisted language learning was developed and pilot-tested with the participation of a broad variety of educators. It is included in Appendix A of this report. Educators are encouraged to use it and modify it, as they see fit. We welcome feedback on it.

3. Year 1 Survey Results: Summary of Needs and Recommendations

Outlined below are the general needs and recommendations identified by language teachers and consultants in Ontario schools in response to the survey on software for language arts that we carried out in Winter 1983 (see Barker et al 1983). Responses identified needs and recommendations in five areas of concern: software, hardware, curriculum, teacher education, and resource staff. While these reflect the particular needs of language arts educators, they also reflect general problems found in nearly every instance of microcomputer use in educational settings.

Software

- Availability: There is little or no good software on the market; this is especially true of French language software.
- Quality: Many available programs are technically faulty; pedagogical standards are frequently unacceptable.
- Evaluation: There are no agreed-upon standards or procedures; release time for teachers is needed for software review.
- Distribution: A centralized cataloguing and clearinghouse is needed; technical support is needed to develop licensing agreements.
- Development: Needs include release time, funding, and coordination; priorities and guidelines for curriculum materials; software design standards; and field testing procedures.

Hardware

- Compatibility: Standards are needed to solve the problem of incompatibility of different hardware and software.
- Availability: Microcomputers, and funding solutions to provide them, are needed.

- Peripherials: Printers are especially needed for effective language applications; disk drives are also required to handle storage of text.

Curriculum

- Standards: Most existing software does not meet current Ministry guidelines for language arts curriculum materials.
- Integration: Guidelines are needed for integration of microcomputers in the language arts program, where their use is warranted, and for use of "software across the curriculum."

Teacher Education

- Pre-service: Language uses of microcomputer need to be integrated into certification programs in Faculties of Education.
- In-Service: Existing "computer literacy" courses need to be extended in number and scope.
- Release Time: More time is needed for teachers to explore various hardware, software, and their uses.

Resource Staff

- Level: More support is needed at all levels of the educational system.
- Type: Staff are needed for professional development, technical support, software development and evaluation, cataloguing, and distribution.
- Specific: A special need exists for resource staff and services to assist educators in French-language schools.

These general needs and recommendations require thorough long-term policies to be developed and coordinated on a Board-wide and province-wide basis. Many of the major policy issues have been identified in recent reports from TV Ontario (1982, 1983), and general recommendations have been formulated in the policy document *Computers in Ontario Education*, the final report of the Provincial Advisory Committee on Computers in Education (Ministry of Education 1983). These important initiatives notwithstanding, immediate practical solutions remain elusive for many language arts teachers interested in microcomputer applications.

4. Year 2 Results: Solutions for Today – Exploiting Available Resources

While waiting for the practical applications that may come from various policy statements, many teachers, schools, and Boards are going ahead and resourcefully making do with the often limited resources at their disposal. This practical experience is a central part of the groundwork for future developments and policies involving educational uses of microcomputers. It is precisely this practical experience that we sought to understand by observing language teachers and students who are using microcomputers in a wide range of situations, from Grade 1 to 13, for English and French as first and second languages. The suggestions put forward below are very much supported by what we have observed. These suggestions are organized around three topics: guiding images, hardware resources, and software resources.

4.1 Guiding Images of Language, Language Learning, and Microcomputers

Perhaps the most limiting feature of the majority of language arts software we have examined is its generally narrow and often superficial view of what constitutes language, language learning, and the role of the microcomputer in educational contexts. While detailed discussion of each of these topics is beyond the scope of this report, it is instructive to consider the images of each that seem to guide the exemplary work of the educators we have observed.

Images of language – In contrast to those views of language that focus separately on subcomponents such as vocabularly, spelling, grammar, paragraph organization, and the like, the view that emerges from exemplary work we have observed has a different, more integrative focus: namely, the purposes for which language is used. Three distinct purposes stand out, though normally all are involved in varying degrees in any natural use of language (see Britton 1970, Canale 1983). These three purposes are:

- language as a tool for thought — or self-directed language, where emphasis is primarily on representing, organizing, and reflecting on ideas and problems (e.g., as in pre-writing activities or in analyzing math problems)
- language as a tool for social interaction — or other-directed language where emphasis is primarily on communication with others and on social relationships with others
- language as a tool for play and art, where emphasis is primarily on self-expression, personal discovery, and personal satisfaction.

Images of language learning — Three crucial elements of language learning are provided for in the exemplary language classrooms:

- instrinsically motivating activities, where the content, format, and purpose of the activity are such that learners want to participate and are led naturally to explore a range of linguistic and non-linguistic information (see Krashen and Terrell 1983, Malone 1981)
- autonomy of the learner, which means that personal learning styles and goals of each learner are considered and, consequently, that learners assume a certain amount of responsibility for both their individual and group learning efforts (see Breen 1982, Holec 1979)
- problem solving at the cognitive, social, and linguistic levels, such that learners are faced with tasks that require them to acquire, use, and reflect upon various knowledge and strategies at each level (see Breen, Candlin and Waters, 1978, Faerch and Kasper 1983).

Images of the microcomputer in educational contexts — In the conclusion of our interim report (Barker et al 1983), we drew attention to the concern expressed by many educators about the role the microcomputer could and should play in educational contexts. Many labels have been invented and applied to the many possible images of computers in education; the roles and labels that we emphasized in that report are, in view of our observations and other research, appropriate and important to re-emphasize here. They are Taylor's (1980) characterizations of the computer as tutor, tool, and, what he calls, "tutee." These roles can be distinguished as follows.

- Tutor — the computer is used to teach or drill the learner, i.e., to play the role of master. In principle, the learner could be allowed to decide what to learn, when to learn it, how to learn it, how well to learn it, how long to take to learn it, with whom to learn it, and how to use it. In practice, the learner (even adult learner), is rarely allowed such control, when tutor software is designed and used.
- Tool — the computer is not intended to teach or drill but to facilitate the learner's (or teacher's) doing something. With the exception of specific authoring templates (e.g., to develop a cloze test), most tool software useful for language learning has not been designed for language learning. Rather, such software has been designed for word processing, for data base creation and management, for controlling video or audio media, for budget or project planning, for simulation, for problem solving, for graphics design, for game construction for music composition, etc.

- Tutee — the learner is required to teach the computer to do something; otherwise, the computer does absolutely nothing. The Logo programming language — designed primarily for use by young children — is perhaps the most widely know example. In principle, the learner has control of his or her learning; in current practice, the learner can become overwhelmed and confused to the point of needing at least to share control with other learners, the teacher, and the computer.

4.2 Hardware: 32 students, one teacher, and one machine

For those who still have an image of individualized computer-assisted instruction with one microcomputer per student, the reality of one microcomputer for an entire class or school may pose an insurmountable obstacle. After all, what can be done that is educationally defensible with such limited access to the machines, especially when there may be so many other demands and obstacles in the curriculum? We observed a variety both of hardware resources and of efforts to cope with the problem of scarce hardware resources.

1. In the least satisfactory arrangement, there was one microcomputer for an entire elementary school. It was made available three days per month to each class. To maximize the use of the computer, one classroom teacher encouraged small group work using a word processing program. He often entered the student's compositions himself to save time. Because both computer-assisted and regular learning activities were carefully integrated and cycled throughout the month, students were prepared to make best use of the computer when it was their turn.

2. The most common setup was the use of a computer lab with six to twelve microcomputers to which a part of the class would go once or twice a week, for an hour or so. This allowed for more ongoing computer-assisted activities, most often focused on writing projects (using word processing software) and on problem-solving and artistic pursuits making use of the graphics capabilities of Logo and of graphics tablets such as the Koala Pad and the Power Pad.

3. Another approach involved language enrichment activities for a small group of three to five ESL children who would come once a week, for half an hour, to a special room with one microcomputer and a teacher. The aim here was to provide a motivating and unthreatening environment to encourage social interaction and opportunities for speaking. Educational games (such as Rocky's Boots), simulations, and reading material most often served to generate interest and content for thought and discussion.

4. Some classes were fortunate to have one or even two microcomputers in the class permanently. Usually, there would be a schedule with one or two children using each machine for half-hour sessions throughout the whole day, while other activities continued in the class. However, several teachers who had initially chosen this arrangement found the continuous noise and distraction difficult to manage; they now prefer a separate microcomputer laboratory where the entire class can work.

5. In a few Boards, microcomputers are arranged in networks to allow for direct communication among students and teachers within a school or

in different schools and towns. Though financially and technically more problematic than other arrangements we observed, this set-up may permit efficient Board-wide use of "electronic bulletin boards," data bases, and software.

What stands out in all these cases is the *social* nature of computer use. Perhaps it is one of the virtues of necessity, but the scarcity of hardware brings with it what all teachers we have observed generally agree is a positive benefit — authentic interaction among the students as they negotiate both for access to the machines and while using them. This view, shared by many has been highlighted by a team of ESL software developers at the University of Victoria, based on their recent experience designing elementary and university level ESL packages:

> "... Our precept is then, that learning or acquisition occurs through this process of communicative interaction, instead of as the direct result of what the program is designed to 'teach'. ... manipulation by a single user alone, in the 'tutorial' mode, is not sufficient for the successful implementation of these lessons in helping students learn English." (Esling, Warbey and Scholefield 1983:7-8).

4.3. Exploiting available software

Given the considerable and still largely justified dissatisfaction with current tutor software designed for language learning, it is not surprising that tool software and tutee software (neither originally designed for language learning) were most often used in the exemplary classrooms in which we observed. Choice of such software seemed to be based on four main considerations;

- reasonable cost
- ease of use
- flexibility of content and methodology.

Discussed briefly below are five types of software that teachers and students seemed to be particularly successful at exploiting for language learning: word processing; authoring systems, templates, and open-ended activities; data base management; simulations and games involving problem solving; and Logo.

From word processing to language and idea processing — Word processing (also known as text editing), is being used in exemplary ways by both teachers and their students.

Use of word processing software by teachers — for their own purposes — is of critical importance. Many curriculum innovations are never successfully implemented because teachers cannot easily integrate them with their personal beliefs and daily practices. This has certainly been the case so far with microcomputers. However, the use of microcomputers for word processing provides an immediate and practical solution to this problem, assuming the availability of a printer. Teachers can begin to use a word processing package on their own to produce a variety of documents —

timetables, worksheets, handouts, tests, letters, etc. Originals can be easily saved, revised, and reprinted as needed.

Once teachers are comfortable with the computer and word processing based on personal use, the prospect of introducing word processing as a component of certain learner activities is no longer so threatening. The variety of possible language learning activities is limited only by the learners' and teachers' imaginations. Among the more exemplary activities we have observed learners engaged in are: drafting and revising their own written text; identifying and discussing ways in which earlier and later drafts differ; seeking (anonymously or publicly) comments from others on written work in progress; creating activities and class tests in which each student contributes his or her own item, and which all students try out; carrying out stylistic modifications to a text for various purposes and audiences; identifying elements of a text which have been intentionally inserted, modified, or deleted; leaving messages for one another and for the teacher; and providing optional on-line explanations of particularly difficult vocabulary and grammar in a reading passage that has been saved on diskette or cassette.

We observed use of a number of word processing packages that can be easily learned and used by both teachers and students. Several are designed specifically for elementary school students. For example, Storywriter, for certain Pet/Commodore microcomputers, was developed by teachers in the Toronto Board of Education and is being introduced in conjunction with the "conferencing" approach to writing adopted by the Board's Language Study Centre. Commercially available packages include the Bank Street Writer, Kidwriter, and Quill (for the Apple II series). Three Icon-based packages that we have not seen in use by young students are the word processing program for the Icon microcomputer (built to specifications set by the Ontario Ministry of Education), Homeword, and the Milliken Word Processor. Several other good packages, frequently used, although not designed specifically for younger students, are Magic Window and AppleWriter IIe for the Apple and Paperclip and WordPro for PET/Commodore microcomputers. Although Applewriter IIe, le Bijou, Megatext, Le Messagier, and Scénario (for Apple II series), and Traitex, Traitex International Scénario (for Pet Commodore)are available in French, many French-language educators have made do with English versions of word processing software.

Since word processing is the most frequently used software in the classrooms we observed, it is worth considering some of its main advantages and disadvantages. The main advantage is that it makes the mechanical aspects of the drafting and revision process simpler to attend to. For example, legibility and neatness are virtually assured; a variety of modifications can easily be made without messy erasures; and new ideas can be quickly inserted or old ones moved around in the text. In principle, this allows more time and thought to be directed towards the higher level cognitive activities of planning, outlining, exploring, discussing, and qualifying one's ideas, i.e., more attention to composition than to transcription (to use the terminology of Frank Smith 1982). A direct and important side effect of this for many learners is increased motivation — not only to write and to revise but also to welcome others to *read* and *talk* about the writing. Penny McCahill (1984), a highschool ESL teacher, has described the con-

structive effects of this increased and authentic social interaction brought about through the combined use of word processing and a "conferencing" approach in her secondary level ESL classes. She has suggested that this exemplifies a useful key theme of all computer-based activities — "teamwork at the keyboard.

One of the most frequent concerns about the use of word processing is the question of typing skills. In all of the classrooms we have observed, not one teacher has found this to be a problem, even in the primary grades, since the task of penmanship is as difficult, if not more so, than "hunting and pecking" on the keyboard. One kindergarten ESL teacher actually turns this to good advantage in teaching the alphabet to Punjabi children. In other situations, older students or the teacher enter the text for a younger student who can then edit it.

Three other disadvantages seem more well-founded. First, most existing word processing software requires the user to devote quite a bit of attention to dealing with the "mechanics" of using a computer; for example, the Bank Street Writer requires a minimum of 13 keystrokes just to save one's work and a minimum of 23 keystrokes to print a standard page (although Applewriter IIe requires only a minimum of three keystrokes for each). As observed by Levin, Boruta, and Vasconcellos (1983), many children spend more time trying to communicate with the machine than communicating their ideas through writing. Second, existing word processing software may encourage attention to superficial, cosmetic aspects of the writing process. For instance, few packages explicitly provide for the planning and outlining stages in the writing process, and all packages make it much easier to delete and insert single characters and words than to move or copy a portion of text, make notes to oneself at certain points in the text, view two pages of text simultaneously, view a preceding or following page, see exactly how the text will look when printed, etc. Finally, most word processing software provides no means for the teacher or learner to keep a record of changes made, keystrokes used, and time spent on writing different portions of the text and on getting the software to carry out certain functions. Anyone interested in the writing process, and not just in the writing product, can appreciate the seriousness of this disavantage.

Together these few advantages and disadvantages illustrate that the concept of "word" processing is too restricted. What we have observed in the most creative uses of word processing software is rather *language processing* and *idea processing* with very rich potential for use across the school curriculum. The recent report of the Ontario Ministry of Education's Computers in Education Project (1983) draws attention to this broader view. And as word processing software becomes more education-minded, powerful, and simple to use, the significance of this view should become even more concrete.

Authoring and open-ended software: No programming required — Authoring systems and templates are powerful tools that simplify the development of certain kinds of software by eliminating the need to use a programming language such as BASIC or Pascal. Authoring systems are generally the more powerful, complex, and costly of the two. Such systems provide comprehensive sets of commands first to set up the overall struc-

ture of a tutorial lesson, drill, or test, then to enter content of the user's choice, and finally to enter prompts, acceptable answers, feedback, any special branching in the lesson (e.g., to allow a learner to skip parts of a lesson that may be too easy or difficult). One well known system that we observed is SuperPILOT for the Apple II series. It allows interesting analysis of learner responses, the creation of different language character sets and color graphics, the use of sound, record-keeping, and interfacing with videodisc. Much less powerful authoring languages also in use in Ontario are IBIS (for the Pet) and Vanilla Pilot (for the Commodore 64). The only authoring system available in French that we have examined is Microscope (for the Apple II Series) from Quebec; while comprehensive, it also is much less powerful than SuperPILOT. A very powerful authoring system that we have not observed in use in Ontario is EnBASIC (cf. Wyatt 1984). These packages require some period of training and practice, but once learned can provide a relatively inexpensive alternative to commercially prepared software, with the very significant advantage of being flexible in content and design.

Templates are much more modest versions of authoring systems, usually with one fixed purpose such as development of a crossword puzzle or cloze test. They allow simple entry and modification of desired content but generally allow no major changes to the activity structure. Two examples for the Apple are Crossword Magic and Quizit.

Open-ended software packages are those whose structure and goals are fixed and, unlike templates, for which optional content is also provided. However, this content is not fixed: teachers and learners have the option of easily entering any desired content. Thus while such packages are much less flexible than authoring systems for designing one's own software, they are also much easier to use. A very recent example of this type in French is the Quebec-produced language program "Expressions simples" in the series Associations 2 (from the company Didacticiels FM). This package allows a teacher either to use vocabulary provided on the diskette or to choose and enter desired vocabulary. First, the students try to identify the meaning of the words, then they see them used in context, and finally,they try to make associations with other words. Some of the more interesting packages of this type use game formats, such as the French programs Tic Tac Jeu and Telejeu (from Logidisque for the Apple II series). The Centre franco-ontarien de ressources pédagogiques in Ottawa has also produced several interesting open-ended educational games (for the Ministry's Icon and for the Apple II series).

Data base management — These packages encourage learners and teachers to gather, organize, modify, and use information about language or anything else. We have observed use of two kinds of data base management. The first kind are the so-called "empty" data base packages which allow any kind of information to be filed and accessed in various ways. Most interesting is their potential as a personal learning resource for students to create and use their own data bases on; for example, specialized vocabulary, favorite idiomatic expressions, science project ideas, or phone numbers of friends. These packages can be used for administrative purposes as well, such as making inventories and keeping track of classroom resource materials (e.g., titles of books or software holdings). ;

We observed learners using a variety of packages for such purposes: Daisy (French and English) and Oracle for the Commodore 64, and Notebook (available through *Window*, a disk-based educational magazine), Data Handler, and PFS: File (Personal File System) for the Apple II series.

The second kind of data base software, less immediately useful for learners but more immediately useful for teachers, is class record-keeping packages. These can simplify some classroom administrative tasks such as recording and calculating student grades. In our observations, the software used for this purpose ranged from the very specific (e.g., "Registre de notes" from Didacticiels FM) to the very general (e.g., Visicalc).

Simulations and adventure games: Problem solving and play — Simulations provide rich and motivating contexts for language (and other) learning by requiring users to assess and respond to changing conditions in the simulated operation of some system. Some very useful simulation software now available involves, for example, maintaining the ecology of a pond, operating a nuclear power plant, buying and selling on the stock exchange, piloting an aircraft, designing machines and products in a factory, and running a lemonade stand. Such software encourages development and use of group interaction skills, global and specific problem solving strategies, and language forms specific to the simulated situations. The preparatory and follow-up activities that such simulations naturally invite are in themselves worthwhile.

Well conceived and produced educational games (e.g., Rocky's Boots, Gertrude's Secrets, Snooper Troops/Profession: détective), also offer useful language learning contexts. One particular genre of game — the adventure game — seems especially rich in potential for language development. This type of game puts groups of learners in real or imaginary worlds in which they must constantly solve cognitive, social, and language problems in order to achieve some larger goal (e.g., complete a quest or solve a mystery). There are several types of adventure games now available. Some, such as In Search of the Most Amazing Thing/sur les traces des. . . , The Dark Crystal, and Le Professeur series, use mostly colour graphics to present the learners with one problem-situation after another. Others such as Zork, Witness, and Suspended use only text in order to encourage the users to rely on their own imaginations without the distraction of often primitive computer graphics. A third type of adventure, known as "fantasy role-playing," lets the learners create their own characters from scratch. From a language learning perspective, the best of these adventures have a built-in sentence parser designed to allow the learners to respond in complete sentences to printed prompts and be reasonably well understood by the program. This combination of reading, interpreting, discussing, and writing provides a rich integrated activity through the medium of the computer.

A major disadvantage of most adventure games is that the content is often inappropriate in many ways (e.g., strong emphasis on violence and frightening scenes, sexual stereotyping, too demanding in terms of language proficiency required, and little topical relevance to any curriculum objectives). As well, many of the games are intended to take months for even adult native speakers to complete. While some students will, of course, persist to the end, others may lose interest. However, with

the help of even a simple authoring language, learners and teachers can plan and develop their own adventures using topics, language, and timelines they find suitable. The availability of adequate adventure game templates, as suggested by Lesgold (1983), could perhaps assist in this effort. Regardless of the means, the goal of learners and teachers creating their own adventure games strikes us as extremely realistic and worthwhile from many educational perspectives

Logo: A tutee approach and valuable prototype — Logo is a programming language that has received a great deal of publicity among educators since its public release about three years ago. One of its main creators, Seymour Papert, a mathematics professor at the Massachusetts Institute of Technology, studied extensively with Piaget and has worked for many years in artificial intelligence research. Logo was intended as a tool for children to discover and explore "powerful" ideas, particularly in mathematics and geometry. It is described in Papert's (1980) book *Mindstorms* as well as in numerous other books and articles since then. Logo is available in various French and English versions for Commodore, Apple, and other microcomputers.

Logo is the best current example of tutee mode software for the microcomputer. It is especially known for its "turtle graphics" in which the "turtle" — a small triangle in the middle of the screen — does absolutely nothing until it is "taught" by the user to draw various objects and shapes (e.g., a hand, a letter of the alphabet, or a square). The user can also "teach" the turtle to "remember" these objects and shapes, to modify them, to animate them, and to combine them into larger objects and shapes (e.g., hands clapping or a neighbourhood scene). Logo also has a very powerful "list-processing" capacity which can be used to do things with language. This capacity is not as well known or as widely used, perhaps because it is harder to work with, less compelling than the graphics mode for many learners, and is limited by the memory size of most of the popular affordable microcomputers on which Logo can run.

Based on our observations, Logo seems attractive for language learning for four main reasons. First, its philosophy of learning coincides with the image of language learning described in Section 4.1. Thus it seeks to provide intrinsically motivating learning contexts, to recognize the importance of learner autonomy and control, and to encourage learning through discovering, analyzing, and resolving problems of many types (including problems in one's own preliminary solutions to larger problems).

Second, as a programming language, Logo is well suited for natural language learning. For example, all programming commands provided in the Logo language are words or phrases in natural language intended to be understood and used by a normal elementary school child; any further commands the user wishes to create can be labeled as he or she chooses. Moreover, as a list-based programming language, Logo encourages users to group together (in lists) those elements that they see as belonging together and to build up larger lists out of smaller ones (e.g., lists of names and action words can be combined into sentences, lists of sentences into paragraphs, and so on). Finally, Logo requires precision in spelling the commands, using them correctly, combining them into larger commands (or procedures), using appropriate "punctuation," analyzing what various

procedures mean, and detecting and correcting one's mistakes. Noteworthy, perhaps, is that this precision is required not for its own sake but rather so that the user can get the computer to do what he or she wants to do. In other words, it is incidental (see Smith 1984).

Third, Logo provides context that generates much language use among learners. Before, during, and after using Logo, we observed that learners were most often involved in discussing and writing out their plans, problems, and reactions. They were involved in negotiating their respective responsibilities, thinking aloud, persuading, explaining, arguing, kidding, criticizing, consoling, congratulating, and many other observable uses of language. One suspects that there were many unobservable uses of language involved as well, such as interpreting and assessing others' views, representing and organizing one's own ideas, and discovering things about oneself and about others.

Finally, Logo provides for a very broad range of interesting language learning activities. One can conceive of at least four distinct types of activities ranging from the very brief and specific to the extended and general. Many creative examples of such applications are described with accompaning Logo programming commands, in the clear informative, monthly *NLX* Newsletter of the National Logo Exchange. The address is included in the list of sources.

- *Rule refinements.* For example, users, might teach the turtle their own rules to handle specific language features (e.g., to form the plural of certain nouns given the singular form, or when to use forms such as *tu* and *vous* or *Pardon* and *Excusez* in Canadian French).
- *Language-use rehearsals.* In a second type of activity the users might try to help the turtle to rehearse a conversation for a particular situation (e.g., to invite a new acquaintance to a birthday party where the situation requires the turtle to greet, respond to a greeting, exchange small talk, invite the person, thank or persuade, give out information, and take leave).
- *Personal information tools.* A third type of activity could encourage students to use Logo's list-processing capability as "empty software" to create and store useful personal information tools. These may be helpful to a learner in organizing easily accessible and modifiable information, such as a glossary for a particular text, rules of thumb, favorite expressions for particular situations, words to songs, or examples of written text formats such as postcards, recipes, or invitations.
- *Extended projects.* Finally, a fourth type of activity could be extended projects that integrate the text, graphics, animation and musical features available in most recent versions of Logo. Thus, for example, groups of students might work on songs, science reports with graphics and animation, or storybooks with text and pictures. Especially valuable for assisting students in these projects are recent Logo utility packages such as Moviemaker (Cohen, Rubincam, Barker, Sinclair, and Watson 1984) which provide help-menus and useful procedures for integrating text, music, and graphics.

Although Logo thus has much potential and promise for assisting in

language learning, for several reasons it is best viewed at present, as only a prototype of intelligent approaches to learning with microcomputers. One reason is that most current versions of Logo still present many frustrations to the user (e.g., limited working memory, no help-menus, cryptic error messages, slow operating speed). Another is that many of the tools Logo seeks to provide are provided in more usable form in other software and hardware (e.g., for graphics, a Koala Pad or software such as Delta Drawing; for text, a simple word processing package such as Storywriter; for text with pictures, a package such as Story Machine or Kidwriter; or for music, software such as the Music Construction Set).

Through our observations, we find that the most important limitation of Logo in its current forms is that it reflects an incomplete view of how learning — certainly language learning — takes place. As pointed out above, Logo explicitly provides for and encourages both learning through doing and learning through one's mistakes. These are two cornerstones of any serious learning theory; they may even suffice for some learners. But perhaps two other cornerstones are not explicitly represented in Logo, at present; learning through observation of an expert and learning through collaboration with an expert. Certainly we have observed learners and teachers who provide such expert performance and collaboration not offered by Logo itself; however, we have also observed enthusiastic learners and teachers who are not always able to provide such expertise — they tend to become very confused and discouraged.

It is for these users and in these situations that a more "intelligent" Logo — an expert turtle, or intelligent tutee turtle, to complement the present tutee turtle — could perhaps be very helpful. This type of Logo could be intelligent in three ways: it would be capable of illustrating solutions to certain problems: and it would collaborate as required with users to help them overcome their difficulties in solving problems. Such intelligent or expert systems are the focus of much recent research and development in both instructional design (e.g., Brown and Burton 1978, Lercher 1983, Sleeman and Brown 1982, Woodruff, Bereiter, and Scardamalia 1983). Logo seems to us to be one of the most promising starting points for bridging the gap between current microcomputer software and more intelligent software.

5. Recommendations and Concluding Comments

Summarized below are the major recommendations emerging from our survey and analysis. Before turning to them, it is worth pointing out two important limitations of these recommendations. One is that they are largely based on the work of experienced and in many ways exceptional language educators. While the practice of these educators is in our view exemplary, instructive, and even inspiring, it is not representative of the current practice and degree of microcomputer experience of the majority of language educators in Ontario. We therefore risk presenting unrealistic recommendations for the majority of language educators. The other limitation is that there are many exemplary language teachers, exemplary software packages, and exemplary uses of microcomputers that we have not observed. For example, we did not observe use of the interesting language software developed in Britain and France (cf. Davies and Higgins 1982, Higgins and Johns 1983), microcomputer networks (cf. Wilton 1983), interactive video and audio (cf. Watt 1984), computerized adaptive testing (cf. Wainer 1983), or the microcomputer used as a research tool (cf. Venezky 1983).

5.1 Summary of major recommendations

Fully aware of the important limitations just noted, we offer four major recommendations concerning the use of microcomputer software for language learning.

Recommendation 1 — Our foremost recommendation is that language educators reflect seriously not only on the educational value of what they are doing *with* microcomputers but also on what they are doing *without* them. Given the hype, pressure, apprehension, and confusion that still characterize too many discussions of educational uses of the microcomputer, it is clear that the machine, learners, educators, and classrooms are not always being used in valuable ways. However, the microcomputer should not become the scapegoat. Rather, it should become a catalyst to stimulate educators to re-examine what they really know and believe — and what their curricula assume — about language and language learning. To be

not just computer literate but also "language literate," and "learner literate," is, in our view, a crucial condition for valuable use of the microcomputer in language learning.

Recommendation 2 — At present, language educators should give priority to tool and tutee uses of software rather than to tutor uses. As pointed out earlier, while in principle, tutor software for microcomputers can be valuable, in practice, its main virtue is too frequently its convenience. It often leaves much to be desired in both educational value and standards. Current tool and tutee software do not guarantee valuable use of microcomputers — they leave much to be desired as well. But because they are more flexible in content and format, they allow learners and educators to use them to suit their language needs and interests. Moreover, because many tool and tutee packages provide very motivating and worthwhile content from other subject areas (e.g., simulations and Logo), a "language and software across the curriculum" approach is encouraged (see Cavanaugh and Styles 1983). Our message to language educators, then, is that they should try to exploit available tool and tutee software that is often not explicitly intended for language learning.

Recommendation 3 — Software evaluation (or software analysis), should be viewed as an ongoing and integral part of software use. Ideally (according to researchers such as Venezky 1983), software should be sufficiently intelligent and flexible to permit evaluation data to be gathered, and the software to be modified, as it is used. For example, a word processing package might record the number of keystrokes, time, and types of help-menu required by a student trying to re-organize a text. At present, such microcomputer software is very rare. A realistic alternative we have observed is to have learners and educators (experienced and inexperienced), note and discuss their reactions to a software package based on a variety of uses on several different occasions. Such evaluative reactions can be systematically encouraged and gathered with the help of an evaluation form and forwarded both to those in the Board responsible for modifying software and to those responsible for producing such software in the first place (see Ragsdale 1983).

Recommendation 4 — Our final recommendation is for Microcomputer Resource Centres within Boards. These Centres now exist in many Boards and perform many valuable functions. We particularly recommend three of those functions as pertinent to the use of microcomputers in language learning. First, those exemplary language educators already using microcomputers can share their experience and expertise with those educators seeking guidance in using microcomputers. Educational use of microcomputers in Ontario is a grassroots movement spearheaded by outstanding classroom teachers rather than by researchers, administrators, and policymakers. These outstanding teachers are likely to be the best means of assuring that other educators receive useful guidance both in inservice training and ongoing support (cf. Anderson 1984). Second, such Centres can assist in co-ordinating Board-wide planning of software and hardware purchases, banks of classroom applications in different subject areas, and evaluative reactions to software and hardware applications. Finally, these Centres can provide a software library and "electronic

bulletin board" that may be consulted, and from which software and messages may be accessed, through a networking system linking individual schools.

5.2. Concluding comments

Our thinking at the close of this project is very nicely summed up in the words and example drawn from four sources.

- The magazine *Science Digest* (August 1983) published an interview with Joseph Weizenbaum, Professor of Computer Science at MIT, co-designer of the first computerized accounting system for banks, and creator of the simulated psychoanalysis program Eliza. Professor Weizenbaum states: "You see, there is a certain kind of thought process that a computer encourages — whether you're balancing checkbooks or simulating some aspect of nature. Let's call it scientific rationality. . . . When the attitude becomes pervasive that science and scientific rationality are the only way of finding out about one's world, we're in terrible trouble" (pp. 94-5).

- In their article "Artificial intelligence, language, and the study of knowledge (in *Cognitive Science* 1.1, 1977) Ira Goldstein and Seymour Papert of MIT raise the following issue: "Suppose that *smart* machines can be constructed. Should educators look to the machines themselves as instruments to make kids smart, or should they rather look at the *principles* which led to the machines becoming more intelligent to see whether these principles could be applied to children?" (p. 114, their emphasis).

- *MacLeans* magazine (November 8, 1982, p. 60b) reported that "the world's most prestigious computer award," the A.M. Turing Award, was presented that week to Professor Stephen Cook of the University of Toronto. Our attention was drawn to three aspects of this event: first, Professor Cook received the award for showing what computers cannot do; second, he does not even use a computer; and third, he never published his findings.

- In his article "Computer promises, computer realities" (in Classroom Computer Learning, March 1984), Lee Otto, who identifies himself only as "an educator," writes of the importance of language skills in the world of tomorrow. He cites from the National Assessment of Educational Progress report *Reading, Thinking and Writing*: "Many students believe they will emerge into an electronic world that will require little reading and less writing. Nothing could be further from the truth" (p. 66).

References

Adams, A., & E. Jones. 1983. *Teaching the humanities in the microelectronic age.* Toronto: Methuen (Open University Press.)

Amarel, M. 1983. The classroom: An instructional setting for teachers, students, and the computer. In A. C. Wilkinson, ed.

Anderson, J. 1984. Teachers key to computer success. *ETS Developments* (Summer 1984): 10-11. (Quarterly publication from Educational Testing Service, Princeton, New Jersey.)

Barker, G., K. MacRury, M. Canale, R. McLean, & R. Ragsdale. 1983 *Survey and analysis of microcomputer software for language arts : Interim report, Year 1.* Toronto: OISE, Centre for Franco-Ontarian Studies.

Blum Cohen, V. 1983. Criteria for the evaluation of microcomputer courseware. *Educational Technology* **23**.1: 9-14.

Bourgain, D. 1983. Enseignement et ordinateurs: Quelle révolution? *Etudes de linguistique appliquée* **50**: 8-30.

Breen, M. 1982. Authenticity in the language classroom. *Bulletin of the Canadian Association of Applied Linguistics* **4**.2: 7-23.

Breen, M., C. N. Candlin, & A. Waters. 1979. Communicative materials design: Some basic principles. *RELC Journal* **10**.2: 1-13. (Journal of the Regional Language Centre of the Southeast Asian Ministers of Education Organization, Singapore.)

Britton, J. 1970. *Language and learning.* Harmondsworth: Penguin.

Brown, J. S., & R. R. Burton. 1978. Diagnostic models for procedural bugs in basic mathematical skills. *Cognitive Science* **2**: 155-192.

CALICO Journal. Computer Assisted Language Learning and Instruction Consortium. 3073 JKHB, Brigham Young University, Provo, Utah 84602. Quarterly Publication.

Canale, M. 1983. On some dimensions of language proficiency. In J. W. Oller, Jr., ed. *Issues in language testing research.* Rowley, Mass.: Newbury House.

Case, R. 1978. A developmentally based theory and technology of instruction. *Review of Educational Research* **48**: 439-463.

Case, R., & C. Bereiter. 1984. From behaviourism to cognitive

behaviourism to cognitive development: Steps in the evolution of instructional design. Occasional paper No. 2. Toronto: OISE, Centre for Applied Cognitive Science.

Cavanaugh, G., & K. Styles. 1983. The classroom computer and language across the curriculum. *Indirections* 8.4: 37-50. (Special issue of the journal of the Ontario Council of Teachers of English on computers and language teaching).

Chandler, D., ed. 1983. *Exploring English with Microcomputers.* London: Council for Educational Technology, 3 Devonshire Street, London WIN 2BA

Chandler, D., & S. Marcus, eds. 1985. *Computers and literacy.* Toronto: Methuen (Open University Press).

Cohen, R., I. Rubincam, G. Barker, N. Sinclair, & N. Watson. 1984. *The development and use of Terrapin Logo based computerized materials in the classroom.* Also *OISE Logo Microworlds Project: Teacher's Guide, Draft Version 1.1* Toronto: OISE, Dept. of Measurement, Evaluation and Computer Applications.

Computers, Reading, and Language Arts. P.O. Box 13247, Oakland, California 94661, U.S.A. Journal.

Davies, G., & J. Higgins. 1982. *Computers, language and language learning.* London: Centre for Information on Language Teaching and Research.

Esling, J., M. Warbey, & J. Scholefield. 1983. ESL courseware — a developmental framework including discussions of university-level and elementary-level programs. In *TEAL Occasional Papers* vol. 8. Victoria, B.C.: University of Victoria.

Faerch, C., & G. Kasper, eds. 1983. *Strategies in interlanguage communication.* London: Longmans.

Geoffrion, L. D., & O. P. Geoffrion, 1983. *Computers and reading instruction.* Reading, Mass.: Addison-Wesley.

Higgins, J., T. Johns. 1983. *Computer-assisted language learning.* London: Addison-Wesley.

Holec, H. 1979. *Autonomie et apprentissage des langues étrangères.* Strasbourg: Council for Cultural Cooperation, Council of Europe.

Holloway, P. 1983. Computers in the language arts classroom. *indirections* 8.4: 30-34.

Hope, G., H. Taylor, J. Pusack. 1984. *Using computers in teaching foreign languages.* Orlando, Fla.: Harcourt Brace, Jovanovich.

Krashen, S., & T. Terrell. 1983. *The natural approach.* London: Pergamon.

Lathrop, A., & B. Goodson. 1983. Evaluation criteria. In *Courseware and the classroom.* Menlo Park, CA: Addison-Wesley.

Lercher, A. 1983. Etude d'un environnement informatique pour des activités linguistiques: état de la recherche. Interim report. Paris: Institut national de recherche pédagogique, Section Information et Enseignement (DP5).

Lesgold, A. M. 1983. A rationale for computer-based reading instruction. In A. C. Wilkinson, ed.

Levin, J. A., M. J. Boruta, & M. T. Vasconcellos. 1983. Microcomputer-based environments for writing: A writer's assistant. In A. C. Wilkinson, ed.

Malone, T. W. 1981. Toward a theory of intrinsically motivating instruction. *Cognitive Science* 4: 333-369.

McCahill, P. 1984. Microcomputer technology in the ESL classroom. *TESL Talk* 15.1: 73-81.

Ministry of Education, Ontario. 1983. *Computers in Ontario Education.* Final report of the provincial advisory committee on computers in education. Toronto: Queen's Park

Papert, S. 1980 *Mindstorms: Children, computers, and powerful ideas.* New York: Basic Books.

Pike, R. 1983. *Evaluating the effectiveness of lessonware prototypes: Some guidelines for use in the development of educational software.* Toronto: Faculty of Education, University of Toronto. Report prepared for the Computers in Education Centre, Ministry of Education, Ontario.

Ragsdale, R. 1983. *Evaluation of microcomputer courseware.* Toronto: OISE Press.

Sleeman, D. H., & J. S. Brown., eds. 1982. *Intelligent tutoring systems.* London: Academic Press.

Smith, F. 1982. *Writing and the writer.* New York: Holt, Rinehart and Winston.

Smith, F. 1984. The promise and threat of microcomputers for language learning. In J. Handscombe, R. A. Orem, & B. P. Taylor, eds. *On TESOL '83: The question of control.* Washington, D.C.: TESOL.

Standiford, S., K. Jaycox, & A. Auten. 1983. *Computers in the English classroom: A primer for teachers.* (ERIC/NCTE, 111 Kenyon road, Urbana, Illinois. ($5.50 U.S.)

Taylor, R. P., ed. 1980. *The computer in the school: Tutor, tool, tutee.* New York: Teachers College Press.

TV Ontario. 1982. *Classroom use of microcomputer.* Microcomputers in Education Series. Toronto: TV Ontario.

TV Ontario. 1983. *Utilisation en classe des micro-ordinateurs.* La série sur les micro-ordinateurs et l'enseignement. Toronto: TV Ontario.

Underwood, J. 1984. *Linguistics, computers and the language teacher.* Rowley, Mass.: Newbury.

Venezky, R. 1983. Evaluating computer-assisted instruction on its own terms. In A. C. Wilkinson, ed.

Wainer, H. 1983. On item response theory and computerized adaptive tests. *The Journal of College Admissions* 28.4: 9-16.

Wilkinson, A. C., ed. 1983.. *Classroom computers and cognitive science.* New York: Academic Press.

Wilton, J. 1983. Structuring learning networks. *ECOO Newsletter (Educational Computing Organization of Ontario)* 4.3: 25-31.

Woodruff, E., C. Bereiter, & M. Scardamalia. 1983. Computers and the composing process: An examination of computer-writer interaction. *ECOO Newsletter* 4.2: 41-45.

Wyatt, D. H. 1984. Computer-assisted teaching and testing of reading and listening.. *Foreign Language Annals (Fall).*

Appendix A

A Computer-Assisted Language Learning Materials Evaluation Form

Acknowledgements

The evaluation form which appears on the following pages was developed based on a review of many existing general-purpose evaluation forms for software. However, particular emphasis on language-related concerns was lacking in these forms and this suggested a need for a more specific evaluation form.

This form draws particularly on four sources: *Evaluation of Microcomputer Courseware* by Ronald Ragsdale, Toronto: OISE Press 1983; *MicroSIFT Evaluator's Guide,* Eugene, Oregon: Northwest Regional Educational Laboratory, 1982; the *Draft Lessonware Evaluation Guide* developed by Gordon S. Wells for the Ontario Ministry of Education 1983; and the *Format for the Description and Evaluation of Materials for Use in Computer-Assisted Language Learning,* developed by Gerry Strei, Nova University, Florida, 1983.

Many people contributed to the design of this form. Prof. Joan Laird, Coordinator, English as a Second Language or Dialect Program, Faculty of Education, University of Toronto, participated in the original discussions. Many helpful suggestions were made by the teachers who attended Prof. Bernard Mohan's course on Computers in Second-Language Learning at the TESOL Summer Institute 1983, held at OISE. Draft versions of the form were reviewed and extensively critiqued by students of the Software Evaluation Course, Fall 1983, at Sheridan College and by the course instructor, Graydon Clipperton. The final draft form was tried out and reviewed by a number of classroom teachers in the Carleton, North York, Ottawa, and Scarborough Board of Education. Their contributions are all gratefully acknowledged, and we hope that they are reflected in some way in this final version.

Suggestions for Use

We have examined all kinds of evaluation forms. Some were the size of a small note pad and asked one question: Would you recommend this package to another teacher? (Yes, No, Maybe). Others were over twelve pages long with accompanying guides of twenty pages. A wide variety of checklists, rating scales, and open-ended response formats were used. The form we have developed is a compromise which has been generally well received in pilot-testing. However a number of important criticisms that were made are still valid, including:

- takes too long to fill out
- takes too long to read
- too thorough; too much detail
- not easy enough to use
- doesn't indicate how to analyse the responses

These comments highlight the difficulty of providing an instrument for all purposes. Each setting will have its own particular evaluation needs and will require methods tailored to them. However, this form may serve as a model for a set of questions and a format for organizing them which tries to deal with the diverse aspects of language learning and the wide variety of types of software

We recommend this form be used by an evaluation team based on observation of the software in "normal" use with the intended or probable user. It is more suitable as a form for summative evaluation purposes where purchasing or selection decisions are being made. However, the questions are also relevant for formative evaluation purposes as well, during the software design process, although the format would likely need revising.

For ease of use, the top page is designed as a Summary Sheet which should provide enough information, at a glance, for a teacher to decide whether she or he is interested in either: 1) using the program or, 2) reading the rest of the form before using or reviewing the package.

The topics and questions are intended to be self-explanatory, although some degree of basic familiarity with computers is assumed. The judgements required are largely subjective. They are intended to guide evaluators to a final recommendation of the program either "as is" or with changes, considering both the intended users and other possible users.

For verification of the basis upon which evaluation judgements are made, there is a special section reserved for a description of the evaluation process itself. Of course, this information is only as useful as the evaluators choose to make it. Ideally this form will be used as part of a co-ordinated software review and distribution process where the evaluation team is generally recognized by teachers throughout the board or school.
One approach then might involve a central resource pool of software and completed evaluation forms from which the summary sheets could be sent to classroom teachers, either on a regular basis or upon request. Based on the summaries, teachers could then decide whether they wanted to request a hands-on review of the program. Where time and resources permit, teachers might also complete the form based on their ongoing classroom use of a package — thus contributing to a developing profile not only of the program's worth but also of alternative ways of using it.

Many informative articles and several very thorough Evaluator's Guides to other software evaluation forms are available (cf. Ragsdale, 1983; Blum Cohen, 1983; Lathrop and Goodson, 1983; Pike, 1983). They provide more detailed information concerning some of the specific items that have been adapted for use in this form. Of these, two are particularly well presented. They can be obtained from:

MicroSIFT
Computer Technology Program
Northwest Regional Educational Laboratory
300 S.W. 6th Ave.
Portland, Oregon
97204
tel. (503) 248-6800

A revised and extended version of the MicroSIFT guide, adapted for local purposes in British Columbia, has been published by PEMC Microware.

PEMC Microware
Provincial Educational Media Centre
7351 Elmbridge Way
Richmond, B.C.
V6X 1B8

Computer-Assisted Language Learning Materials Evaluation Form

Ontario Institute for Studies in Education & Faculty of Education U. of Toronto

---

SUMMARY SHEET

A. General Description

Catalogue Info: _____

Package Title: _____ Cost: _____

Topic/Theme: _____

Language(s): _____ Author(s) _____

Publisher & Address: _____

_____

Distributor: _____

_____

Is documentation and /or support material included?

[ ] Yes [ ] No

Is the package in the public domain?

[ ]Yes [ ] No

State publisher's conditions regarding back-up copies and/or licensing:

B. Hardware and Software Required

Computer Model(s): _____ Memory: _____ K.

Peripherals and Interface Devices (List all necessary) _____

_____

Software: Format: [ ] Diskette [ ] Cartridge [ ] Cassette Tape [ ]

Other _____

Type and version of programming language and DOS, (if known) _____

C. Intended Users

[ ] Students: School/Grade Level(s): _____

Second-language Level: [ ] N/A [ ]Beginning [ ] Intermediate [ ]Advanced

[ ] Teachers [ ] Courseware Designers [ ] Other (specify) _____

[ ] For individual use [ ] For small group use [ ] Usable with either

D. Package Overview Summarize the Package.

E. Recommendation (To be completed as the *final step* of the evaluation, based on the detailed questions on the following pages. For information on the evaluation process see Part J, page 34)

I/we would use or recommend use of this package:

	Yes	*No*
1. *For the intended users (stated or inferred):*		
a) with little or no change	[ ]	[ ]
b) only if certain changes were made (refer to Part 1, p. 33)	[ ]	[ ]
2. *For other users:*		
a) with little or no change	[ ]	[ ]
b) only if certain changes were made (refer to Part 1, p. 33)	[ ]	[ ]

F. Package Objectives and Learning Outcomes

1. Indicate the package objectives. Are they stated in the Package?
[ ]Yes [ ]No

2. a. How well does the package fulfil these objectives?
[ ] not applicable. [ ] completely [ ] somewhat [ ] not at all

b. Explain:

3. For what other objectives or expected learnings may the package be suitable?

Curriculum Goals

4. Identify the specific curriculum guidelines, areas or topics to which this package may be related:

Type of Software:

5. *Circle* the main type and check others if applicable.

[ ] Tutorial	[ ] Logo Program
[ ] Drill and Practice	[ ] Programming Language
[ ] Situational Activity	[ ] Authoring System/Template
[ ] Word Processing	[ ] Database
[ ] Composition Aid	[ ] Learning/class management
[ ] Game	[ ] Test Assessment
[ ] Problem-Solving	[ ] Utility
[ ] Simulation	[ ] Other (describe) _____

Prerequisite Knowledge and Skills

6. Indicate necessary prerequisite skills and knowledge:

G. *Content*

The following aspects relate to the intended users indicated in Part C.

(For each item, *circle* one star from the rating scale: SA = Strongly Agree, A = Agree, NoOp = No Opinion, D = Disagree, SD = Strongly Disagree, NA = Not Applicable)

Language

1.	SA	A	NoOp	D	SD	NA	The package serves as a good model of language for the students. Consider the following:
	*	*	*	*	*	*	Grammar
	*	*	*	*	*	*	Vocabulary
	*	*	*	*	*	*	Language use situations
	*	*	*	*	*	*	Organization and coherence
	*	*	*	*	*	*	Language style (e.g. formal/ informal
	*	*	*	*	*	*	Other (specify) _____

2a. The package contains errors in : Check all applicable)

[ ] Spelling [ ] Grammar [ ] Punctuation [ ] Capitalization
[ ] Accents [ ]Other

b. Specify the most frequent or important errors:

	SA	A	No-Op	D	SD	NA	
3.							The package encourages the user to interact actively with it by:
	*	*	*	*	*	*	reading
	*	*	*	*	*	*	writing
	*	*	*	*	*	*	talking
	*	*	*	*	*	*	listening
4.	SA	A	NoOp	D	SD	NA	The language of the package is:
	*	*	*	*	*	*	consistent with the grade level (see Part C)
	*	*	*	*	*	*	consistent with the second-language level (" ")
	*	*	*	*	*	*	appropriate in terms of required reading ability
5a.	SA	A	NoOp	D	SD	NA	The package deals adequately with a variety of unexpected user responses. Consider each of:
	*	*	*	*	*	*	Grammar
	*	*	*	*	*	*	Vocabulary
	*	*	*	*	*	*	Word Endings
	*	*	*	*	*	*	Spelling
	*	*	*	*	*	*	Punctuation
	*	*	*	*	*	*	Capitalization
	*	*	*	*	*	*	Accents

5b. Describe how incorrect or unexpected responses are handled by the program:

6. Summarize your opinion of the language content of the package:

Cultural References and General Knowledge

7.	SA	A	NoOp	D	SD	NA	In the context of the package, the specific cultural references are:
	*	*	*	*	*	*	accurate
	*	*	*	*	*	*	clearly presented
	*	*	*	*	*	*	acceptable
	*	*	*	*	*	*	free of racial, ethnic, sexual, religious, or social class stereotypes.

8.	SA	A	NoOp	D	SD	NA	In the context of the package, the specific items of general knowledge are:
	*	*	*	*	*	*	accurate
	*	*	*	*	*	*	clearly presented
	*	*	*	*	*	*	acceptable

H. Package Design

Instructional Quality

1.	SA	A	NoOp	D	SD	NA	CONTENT
	*	*	*	*	*	*	The content has educational value.
	*	*	*	*	*	*	Presentation of content is clear and logical.
	*	*	*	*	*	*	The content can be easily changed as needed.
	*	*	*	*	*	*	Suitable diagnostic capabilities are provided.
	*	*	*	*	*	*	The instructions are clear to the user.
	*	*	*	*	*	*	Adequate examples are provided.
	*	*	*	*	*	*	Package can be easily integrated with other learning materials, on or off the computer.

2.	SA	A	NoOp	D	SD	NA	USE
	*	*	*	*	*	*	The package is simple to use.
	*	*	*	*	*	*	The package encourages creativity of user.
	*	*	*	*	*	*	Interaction with other people is encouraged.
	*	*	*	*	*	*	A "Help" function and/or menu is provided.
	*	*	*	*	*	*	The user can control rate or presentation.
	*	*	*	*	*	*	The user can control sequence of activities.
	*	*	*	*	*	*	Feedback is appropriate.
	*	*	*	*	*	*	Testing/evaluation capabilities are appropriate.
	*	*	*	*	*	*	Record-keeping capabilities are appropriate.
	*	*	*	*	*	*	Graphics are use appropriately.
	*	*	*	*	*	*	Sound is used appropriate.
	*	*	*	*	*	*	Work can be printed out easily from package.

Technical Quality

3.	SA	A	NoOp	D	SD	NA	
	*	*	*	*	*	*	The package is reliable in normal use.
	*	*	*	*	*	*	Intended users can easily operate the package.
	*	*	*	*	*	*	Teachers can easily operate the package.
	*	*	*	*	*	*	Program and support materials are well-packaged.
	*	*	*	*	*	*	Screen displays are effective.
	*	*	*	*	*	*	The cost of the package is appropriate.

4.	SA	A	NoOp	D	SD	NA	The package effectively uses:
	*	*	*	*	*	*	Interface devices (joystick, mouse, graphics pad)
	*	*	*	*	*	*	Video-tape and/or videodisc
	*	*	*	*	*	*	Oral interaction (e.g., on cassette; synthesized or digitilized voice).

Documentation and Support Materials

5.	SA	A	NoOp	D	SD	NA	
	*	*	*	*	*	*	Teacher support materials are comprehensive.
	*	*	*	*	*	*	Teacher support materials are effective.
	*	*	*	*	*	*	Student support materials are comprehensive.
	*	*	*	*	*	*	Student support materials are effective.

6a. Check all the kinds of materials included in the package:

[ ] teacher manuals
[ ] student manuals
[ ] user/technical manual
[ ] lesson plans
[ ] other _____
[ ] prerequisite activities
[ ] follow-up activities
[ ] worksheets
[ ] demonstration on disk

b. Are materials required other than those provided in the package?
[ ] Yes [ ] No.

If yes, describe:

I Summary

Summarize the package by describing:

1. Major Strengths:

2. Major Weaknesses:

3. Potential Uses:

4. Recommended Changes:

PLEASE COMPLETE NEXT PAGE.

J. Evaluation Information

1. Evaluated by: Date:

Address:

Phone:

Position(s):

2. Relevant background of evaluator(s):

3. Evaluation process included observation of use by intended users:
[ ] Yes [ ] No

4. Describe the evaluation process and summarize the observations made:

PLEASE INDICATE YOUR RECOMMENDATION IN PART E ON PAGE 28. THANK-YOU.

Appendix B

Suggested Sources of Software

Since the interim report of this project was published in 1983, the quality of educational software has improved markedly. Many new producers and programs have appeared and been well publicized in educational computing magazines. The following list of suppliers is fairly comprehensive, but certainly not exhaustive. It was revised in October, 1985

Acorn Computer 14 Woodborough Road, Guelph, Ont. N1G 3K5	– Acorn – catalogue of BBC Acorn materials – includes language programs, simulations
Aurora Software P.O. Box 1394 Haileybury, Ont. P0J 1K0 (705) 672-5517	– PET/CBM Commodore 64 – includes "Storywriter" and "Textmaster" – includes "Kapuskasing French" programs
BIPACS (Bilingual Publications and Computer Services) 33 West Walnut St. Long Beach, NY. USA 11561	– Apple (some use digitilized voice as well) – various foreign language programs – ESL, French and Spanish
Carleton Board of Education Computer P.L.A.C.E., 133 Greenbank Road, Nepean, Ont K2H 6L3	– Apple – have compiled an extensive catalogue of educational software for the Apple – printed catalogue available at cost.

Centre des manuels scolaires 277, rue Hutchings Winnipeg, Manitoba R2X 2R4	— Apple, Commodore — teacher developed French language software for several subjects — Commodore catalogue #977207 ($2.95) — Apple catalogue #97725 ($2.50)
CEC (Centre Educatif et Culturel) 8101, Boul. Métropolitan, Montréal Québec H1J 1J9 (514) 351-6010	— Apple, Vic-20, TRS-80 IBM-PC — various simulations, games and utilities — includes programs from Logidisque: "Caraibes" "Arsène Larcin," "Le Jeu du Chomeur"
CLEF (Computer-Assisted Learning Exercises for French) Language Laboratories, University College, University of Western Ontario London, Ont., N6A 3K7 (519) 679-2554/6136	— Commodore, IBM-PC — series of French grammar exercises — 62 lessons, two lessons per diskette — for IBM PC version (with color), D. M. Paramskas, Dept. of Languages and Literature, University of Guelph, Guelph, Ont. N1G 2W1 (519) 824-4120 ext. 3883
CFORP (Centre franco-Ontarien de Ressources Pédagogiques) 339, Rue Wilbrod Ottawa, Ont. K1N 6M4 (613) 238-7957	— Apple, Commodore 64, CBM-4032, ICON —developed a wide variety of programs in French — distribute others in French which have been created by educators in Ontario French-language schools
Commodore c/o Yvon Husereau 625 Président Kennedy, Montréal, Québec, H3A 1K2	— Commodore 64, CBM, PET — for list of distributors of Commodore Terrapin Logo Française
Computerre/Le Professeur C.P. 782 St. Laurent, Québec H4L 4W2 (514) 747-9130	— Apple — French translationsof adventure games — simulations and Math series in French — multi-lingual template and word processor — distributed by CFORP also
CONDUIT Litus Software 3952 Chesswood Drive Downsview, Ont. M3J 2W6	— Apple, Pet, TRS-80, Atari — ESL, Spanish and French — grammar tutorials and drills, simulations — a natural language processor "Dasher"

Didatech Software 2301-1150 Jervis St. Vancouver, B.C. V6E 2C8 (604) 687-3468	– Apple – English word search "Fay the Word Hunter" – also "Fay the Math Hunter," problem solving
Didacticiels FM 1113 Desnoyers, Laval, Québec H7C 1Y6 (514) 324-0712	– Apple, Commodore 64, IBM-PC Max –open-ended programs, content can be changed – includes "Associations," "Racines," "Interrogations," "Différenciations," "Homophones," record-keeping program "Registre de Notes" – "le Messagier," French Word-processor for young children – "Logo en Français" for C-64 and French Logo manual – Traitex 64 and Traitex International (see visiontronique listing) – French versions of Spinnaker software
Didakteck 2669 Chemin Sainte-Foy Sainte-Foy, Québec G1V 1V3 1-800-463-5369	–Apple, Commodore 64, IBM Privaticiels "Youmi" –also available through Didacticiels FM
Editions Champlain 107 Church Street Toronto, Ontario (416) 364-4345	– Apple, Commodore 64, IBM-PC – distributors of several French software producers: Logidisque, Didaktek, Vifi, Vision, D.C. Heath, Chiasson
Edu-Ware Services Inc. 2222 Sherman Way, Suite 102 Canoga Park, CA 91303	– Apple, Commodore –Various programs
FL2 Software Publications 49 Gillespie Crescent, Ottawa, Ontario. K1V 0C1	– Apple – series of FSL grammar/structure exercises – uses SuperPILOT, graphics, record-keeping
Gessler Educational Software Suite 10A, 900 Broadway, New York NY, USA 10003	–Apple, Commodore, Atari, IBM-PC, TRS 80 – multilingual word processing program, "Type-writer"; English and 10 foreign languages – CLEF series of French grammar exercises

Guérin 4501, rue Drolet, Montreal, Québec H2T 2G2 (514) 842-3481	— Apple, IBM-PC Commodore 64, TRS 80 — authoring system in French, "Micro-scope" — programs developed using Micro-scope in various subjects in French, FSL, and ESL
Litus Software 3952 Chesswood Drive Downsview, Ont. M3J 2W6	— distributor of CONDUIT, Microcomputer workshops, Milliken, Sunburst, and Spinnaker programs
Logéduc Inc. 1117 rue Ste-Catherine ouest Bureau 601, Montréal, Québec H3B 1H9 (514)849-2231	—IBM-PC, MICRAL 90/50 (16 bits) —French authoring system "EGO", Enseignement Guidé par Ordinateur — dialogue format, "intelligent" capability — considers user's response in dialogue
MicroMonde C.P. 557 Orléans, Ont. K1S 1C9	Apple —booklet for introducing Apple Logo, — "Mes Premiers Pas Avec La Tortue"
National Logo Exchange Newsletter P.O. Box 5341 Charlottesville, VA USA 22905	— all versions of Logo discussed — excellent practical examples of many logo applications including language learning activities — monthly September — May $25 (US) per year
Ontario Educational Software Service, TV Ontario, Client Services/Educational Software Box 200, Station Q Toronto, Ont. M4T 2T1 (416) 484-2665	— Apple, Commodore, ICON — distributes learning materials developed under Ontario Ministry of Education, Exemplary Lessonware Project for Ontario school boards and approved educational institutions — others must purchase directly from developers.
PROMPT (Program for Reading Comprehension — Teacher Controlled) D. Mydlarski, Dept. of French, Spanish, Italian, University of Calgary, Calgary, Alberta T2N 1N4 (403) 284-5300 ext. 5303	— IBM-PC — an easy to use authoring template — can enter any text up to 48 lines — multiple choice or fill-in-the-blanks — easy to edit text, questions, responses —French, Italian, Spanish, German

Puce (Programmation Utilisé et Conçue pour l'Enseignement) 342, Rue Sherbrooke est, Montréal Québec (514) 288-5506	— Apple, IBM-PC, Commodore 64 — programs in various subject areas in French — includes an intro to Logo, adventure games, simulations, record-keeping programs — distributed by Didacticiels FM
Regents/ALA Company 2 Park Avenue New York, NY 10016 (212) 889-2780	—Apple ESL CAI series by American Language Academy — colour graphics, record-keeping — Quizit, a mini-authoring program
Québit C.P. 292, Montréal Québec H2C 3G2 (514)668-5701	— Apple "Megatexte", French word processor —"MEGAMOTS," dictionary and spelling checker for "Megatext" — "Spatiotest," game software with math, French, and English questions; includes authoring system to add items
Scarborough Board of Education Computers in Education Dept. 140 Borough Drive Scarborough, Ontario M1P 4N6	— Apple — designed and developed a variety of programs — includes "Puzzler" for reading strategies
Scholastic Software 123 Newkirk Road Richmond Hill, Ontario L4C 3G5 (416) 883-5300	— Apple, Commodore, IBM-PC — Various programs including "Story Tree"
Sierra On-Line Inc. Sierra On-Line Bldg., Coarsegold, USA 93614	— Apple, Commodore, IBM French adventure game, "Mystery — House" — English word processor, "Homeword," uses icons
Software Research Corporation Discovery Park University of Victoria P.P. Box 1700, Victoria, B.C. V8W 2Y2 (604) 477-7246	— Apple —ESL specialists, integrate voice on cassette — games: "My ESL Baseball Program" — simulations: "George Appleby's Grocery Store" — word processor for children: "Little Wordsworth"
Sunburst Communications 39 Washington Avenue Pleasantville, New York 10570	— Apple Commodore, IBM-PC — wide variety of good quality programs

Systèms d'ordinateurs LOGO 9960, Côte-de-Liesse, Lachine, Québec H8T 1A1 (514) 631-7081	– Apple – Apple Logo Française
VIFI Sogiciel 955, rue Amherst, Montréal, Québec H2L 3K4 (514) 523-1182	– Apple French stories, games and simulations some open-ended, students can change content, – "Le rédacteur", French word processor
Visiontronique 218, rue Alexandre Sherbrooke, Québec J1H 4S7	– Commodore 64, CBM – French word processor, "Traitex" – Traitex International with expandable dictionary spelling checker and text analysis – distributed by Didacticiels FM – French/English data base, "Daisy"
Wida Software 2 Nichols Gardens, London, England W5 5HY	– Apple, Sinclair ESL and foreign language programs – includes "Storyboard," "Clozemaster," "Apfeldeutsch" (has audio cassettes)

Innovative language teachers are using the microcomputer with their students at all grade levels. Microcomputer Software for Language Arts tells how these teachers and their students are exploiting the microcomputer for language learning.

A useful software evaluation form is included, along with a list of the mainly Canadian sources of software that the authors have examined.

ISBN: 0-7744-5075-4
OISE Press/The Ontario Institute for Studies in Education